AMENDMENTS TO THE CLAIMS

Applicant submits below a complete listing of the current claims, including marked-up claims with insertions indicated by underlining and deletions indicated by strikeouts and/or double bracketing. This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of the Claims

1. (Currently Amended) A method of operating a computer system to validate the data ~~stored~~ defining a design of an integrated circuit, the method comprising:

storing said data in a plurality of data files in a database each of said data files having an associated file type and being arranged in a plurality of data stores in said data base, each of said data stores relating to a different aspect of said design, wherein at least one of said data files is a data dependent file containing data dependent on data in one or more other of said data files; ~~of said data store, said method comprising the steps of:~~

selecting a file locator which is associated with a respective one of said data stores in said data base relating to a respective one of said aspects of the design;

using said selected file locator [[for]] to identify[[ing]] a first data dependent file from said data files and to identify[[ing]] one or more other of said data files upon which said first file is dependent;

for each of said identified files, selecting a first file reader associated with the file type of the identified file;

using said selected first file reader, determining a predetermined parameter of said identified file;

comparing the predetermined parameter from the first file with that from ~~the or~~ each other file; and

~~responsive to said comparison step~~, providing an output signal for each data file, in response to the comparing, indicating whether the data file is valid for use in manufacture of said integrated circuit.

2. (Currently Amended) A method according to claim 1, wherein ~~said step of~~ identifying files comprises ~~the steps of~~:

locating, via said file locator, files which contain dependency information;

for each located file, selecting a second file reader associated with the file type of the located file; and

via said second file reader identifying said first dependent file and each other file on which the first file depends.

3. (Currently Amended) A method according to claim 1 further comprising ~~the steps of~~:

for each data store generating a list therein containing an entry for each first dependent file in the data store, said entry including a first record having details of the first dependent file.

4. (Original) A method according to claim 3, wherein each entry in said list further includes a further record for each other identified file upon which the dependent file depends.

5. (Currently Amended) A method according to claim 1 further comprising ~~the steps of~~:

selecting the file locator from a file locator means which contains a plurality of file locators; and

selecting a file reader from file reader means which contain a plurality of file readers.

6. (Previously Presented) A method according to claim 1, wherein said predetermined parameter comprises the date on which the data file was last modified.

7. (Previously Presented) A method according to claim 1, wherein said predetermined parameter is a UNIX date stamp.

8. (Currently Amended) A method according to claim 1 further comprising ~~the steps of~~:

identifying every said first dependent file in said data store.

9. (Previously Presented) A method according to claim 1, wherein said data store is a data base library.

10. (Currently Amended) A computer system arranged to validate data ~~stored~~ defining a design of an integrated circuit, the computer system comprising:

storing means for storing said data in a plurality of data files in a data base each of said data files having an associated file type and being arranged in a plurality of data stores in said data base, each of said data stores relating to a different aspect of said design, wherein at least one of said data files is a data dependent file containing data dependent on data in one or more other of said data files; ~~of said data base, said system comprising:~~

a plurality of file locators each associated with a respective one of said data stores in said data base relating to a respective one of said aspects of the design, and arranged to identify a first data dependent file from said data files in said associated data store and one or more other of said data files in said data base upon which said first file is dependent;

a plurality of file readers each associated with a respective file type and each arranged to determine a predetermined parameter for at least one identified file having that associated file type;

comparison means arranged to compare the predetermined parameter determined for said first file, with the predetermined parameter determined for each other file; and

output means responsive to said comparison means and having an output which indicates whether said first file is valid for use in manufacture of said integrated circuit.

11. (Original) A computer system according to claim 10, said system further comprising:

in each data store, at least one file which can be located and which contains dependency information which enables dependent files and said other files in the data store to be identified; and

means provided to locate said located file.

12. (Original) A computer system according to claim 11, further comprising:

a file reader associated with the located file which is adapted to provide a list in the data store, said list having an entry for each dependent file having details contained in the located file and including a record in said entry for said dependent file together with a further record for each other file upon which the dependent file depends.

13. (Previously Presented) A computer system according to claim 10, further comprising:

file locator means containing the plurality of file locators; and

file reader means containing the plurality of file readers.

14. (Previously Presented) A computer system according to claim 10, wherein said predetermined parameter comprises the date on which the date file was last modified.

15. (Previously Presented) A computer system according to claim 10, wherein said predetermined parameter is a UNIX date stamp.

16. (Previously Presented) A computer system according to claim 10, wherein said data store is a data base library.

17. (Currently Amended) A computer program product comprising a computer readable medium having thereon:

computer program code means, when said program is loaded to make the computer perform a method to validate data defining a design of an integrated circuit, the method comprising:

storing said data ~~stored~~ in a plurality of data files in a database each of said data files having an associated file type and being arranged in a plurality of data stores in said data base, each of said data stores relating to a different aspect of said design, wherein at least one of said data files is a data dependent file containing data dependent on data in one or more other of said data files; ~~of said data store, said method comprising the steps of:~~

selecting a file locator which is associated with a respective one of said data stores in said data base relating to a respective one of said aspects of the design;

using said selected file locator [[for]] to identif y[[ing]] a first data d ependent file and identifying one or more other files on which said first file is dependent;

for each of said identified files, selecting a first file reader associated with the file type of the identified file;

using said selected first file reader, determining a predetermined parameter of said identified file;

comparing the predetermined parameter from the first file with that from ~~the or~~ each other file; and

~~responsive to said comparison step~~, providing an output signal for each data file, in response to the comparing, indicating whether the data file is valid for use in manufacture of said integrated circuit.

18. (Currently Amended) A computer readable medium, having a program recorded thereon, where the program is to make the computer perform a method to validate data defining a design of an integrated circuit, the method comprising:

storing said data ~~stored~~ in a plurality of data files in a database each of said data files having an associated file type and being arranged in a plurality of data stores in said data base, each of said data stores relating to a different aspect of said design, wherein at least one of said data files is a

data dependent file containing data dependent on data in one or more other of said data files; ~~of said data store, said method comprising the steps of:~~

selecting a file locator which is associated with a respective one of said data stores in said data base relating to a respective one of said aspects of the design;

using said selected file locator [[for]] to identify[[ing]] a first data dependent file from said data files and to identify[[ing]] one or more other of said data files upon which said first file is dependent;

for each of said identified files, selecting a first file reader associated with the file type of the identified file;

using said selected first file reader, determining a predetermined parameter of said identified file;

comparing the predetermined parameter from the first file with that from ~~the or~~ each other file; and

~~responsive to said comparison step,~~ providing an output signal for each data file, in response to the comparing, indicating whether the data file is valid for use in manufacture of said integrated circuit.

19. (Currently Amended) The computer program product of claim 17, wherein ~~said step of~~ identifying files comprises ~~steps of~~:

locating, via said file locator, files which contain dependency information;

for each located file, selecting a second file reader associated with the file type of the located file; and

via said second file reader identifying said first dependent file and each other file on which the first file depends.

20. (Currently Amended) The computer program product of claim 17, wherein the method further comprises ~~steps of~~:

for each data store generating a list therein containing an entry for each first dependent file in the data store, said entry including a first record having details of the first dependent file.

21. (Previously Presented) The computer program product of claim 20, wherein each entry in said list further includes a further record for each other identified file upon which the dependent file depends.

22. (Currently Amended) The computer program product of claim 17, wherein the method further comprises ~~steps of~~:

selecting the file locator from a file locator means which contains a plurality of file locators; and

selecting a file reader from file reader means which contain a plurality of file readers.

23. (Previously Presented) The computer program product of claim 17, wherein said predetermined parameter comprises the date on which the data file was last modified.

24. (Previously Presented) The computer program product of claim 17, wherein said predetermined parameter is a UNIX date stamp.

25. (Currently Amended) The computer program product of claim 17, wherein the method further comprises ~~a step of~~:

identifying every said first dependent file in said data store.

26. (Previously Presented) The computer program product of claim 17, wherein said data store is a data base library.

27. (Currently Amended) The computer readable medium of claim 18, wherein said step of identifying files comprises ~~steps of~~:

locating, via said file locator, files which contain dependency information;

for each located file, selecting a second file reader associated with the file type of the located file; and

via said second file reader identifying said first dependent file and each other file on which the first file depends.

28. (Currently Amended) The computer readable medium of claim 18, wherein the method further comprises ~~steps of~~:

for each data store generating a list therein containing an entry for each first dependent file in the data store, said entry including a first record having details of the first dependent file.

29. (Previously Presented) The computer readable medium of claim 28, wherein each entry in said list further includes a further record for each other identified file upon which the dependent file depends.

30. (Currently Amended) The computer readable medium of claim 18, wherein the method further comprises ~~steps of~~:

selecting the file locator from a file locator means which contains a plurality of file locators; and

selecting a file reader from file reader means which contain a plurality of file readers.

31. (Previously Presented) The computer readable medium of claim 18, wherein said predetermined parameter comprises the date on which the data file was last modified.

32. (Previously Presented) The computer readable medium of claim 18, wherein said predetermined parameter is a UNIX date stamp.

33. (Currently Amended) The computer readable medium of claim 18, wherein the method further comprises ~~a step of~~:

identifying every said first dependent file in said data store.

34. (Previously Presented) The computer readable medium of claim 18, wherein said data store is a data base library.

35. (Previously Presented) The method of claim 1, wherein each of the plurality of data stores is of a different type and wherein the step of selecting a file locator further comprises:

selecting a different file locator for each store of a different type.

36. (Previously Presented) The computer system of claim 10, wherein each of the plurality of data stores is of a different type.

37. (Previously Presented) The computer program product of claim 17, wherein each of the plurality of data stores is of a different type and wherein the step of selecting a file locator further comprises:

selecting a different file locator for each store of a different type.

38. (Previously Presented) The computer readable medium of claim 18, wherein each of the plurality of data stores is of a different type and wherein the step of selecting a file locator further comprises:

selecting a different file locator for each store of a different type.